

WHAT IS CLAIMED IS:

1. A method for receiving event notification in a network, comprising:
 - subscribing to a first event source;
 - receiving zero or more event messages; and
 - tearing down the event subscription.
2. The method as recited in claim 1 further comprising:
 - receiving at least two event messages each comprising a sequence number and a time stamp from the first event source when events occur; and
 - determining the order of events within the first event source on the basis of the sequence numbers within the event messages.
3. The method as recited in claim 2 further comprising:
 - subscribing to a second event source;
 - receiving at least two event messages each comprising a sequence number and a time stamp from the second event source when second events occur;
 - determining the order of events within the second event source on the basis of the sequence number within the event messages from the second event source; and
 - ordering the events from the second event source with respect to the first event source on the basis of the timestamp within each of the first event messages and the second event messages.
4. The method as recited in claim 1 wherein the event messages are described with a Type Description Language.
5. The method as recited in claim 1 wherein the event messages are delivered as SOAP messages.
6. The method as recited in claim 1 wherein the event messages can convey both absolute (initial) and relative (delta) values.

7. The method as recited in claim 1 wherein the event source defines the events it raises as a name-type pair.

5 8. The method as recited in claim 1 wherein the event source and event sink are identified using standard types IEventSource and IeventSink, respectively.

10 9. The method as recited in claim 1 wherein the event source supports filtering of events it raises.

10 10. The method as recited in claim 1 wherein a subscriber can establish an event filter as part of the initial subscription.

15 11. The method as recited in claim 1 wherein a subscriber can update the event filter established as part of the initial subscription.

20 12. The method as recited in claim 4 wherein the Type Description Language comprises a markup language.

25 13. The method as recited in claim 1 wherein the first event source messages are one-way messages.

14. The method as recited in claim 1 wherein the subscription is made to the first event source by way of an intermediary.

25 15. The method as recited in claim 1 wherein the event callback goes through an intermediary.

30 16. The method as recited in claim 1 wherein the subscriptions are defined in a type description language.

17. The method as recited in claim 11 wherein the type description comprises a one to one mapping to an extensible markup language.

5 19. The method as recited in claim 1 wherein the first event source is an object on a digital device.

10 20. The method as recited in claim 1 comprising setting a lease term after the expiration of which the first event source discontinues the transmission of event messages.

15 21. The method as recited in claim 20 comprising sending a renewal message to renew the lease term.

22. The method as recited in claim 1 wherein the network is an intranet.

15 23. The method as recited in claim 1 wherein the network is the Internet.

20 24. A computer-readable medium bearing computer-readable instructions for carrying out the steps recited in claim 1.

25 25. A distributed system comprising:

 a first digital device;

 a second digital device capable of communicating with the first digital device by way of a computer network,

25 said first digital device subscribing to a first event source operating on the second digital device whereby the first digital device receives event notification messages each comprising a sequence number and a time stamp from the first event source when events occur.

30 26. The system as recited in claim 25 further comprising an intermediary device in communication with the first digital and second digital device whereby event notification

messages are routed to the intermediary device and thereafter forwarded to the first digital device.

5 27. The system as recited in claim 25 wherein the messages are constructed in a type

description language.

10 28. The system as recited in claim 27 wherein the type description language has a one to
one mapping to an extensible markup language.

15 29. The system as recited in claim 25 wherein the first digital device determines the
order that events occurred on the second digital device by way of the sequence number.

30. The system as recited in claim 25 wherein the event messages are one-way messages.

15 31. The system as recited in claim 25 wherein the first and second digital device are
coupled to an intranet.

20 32. The system as recited in claim 26 wherein the first and second digital device are
coupled to the Intranet.

25 33. A method for using to services in a computer network, comprising:
subscribing to an event on a first digital device;
receiving a indication in a type description language comprising a timestamp and
sequence number that the event has occurred on the first digital device; and
requesting a service to be performed by the first digital device after receiving the
indication that the event has occurred.

34. The method as recited in claim 33 wherein the type description language has a one to
one mapping to an extensible markup language.

35. The method as recited in claim 33 wherein the subscription comprises a lease term after which an event message will not be received from the first digital device.

5 36. The method as recited in claim 33 comprising sending a renewal message to the first digital device whereby the lease term is extended.